Guidance and Control, For Small AUVs Using DGPS and Doppler Aided Inertial Underwater Navigation

Anthony J. Healey
Center for AUV Research
Naval Postgraduate School
Monterey, CA

Abstract-**This paper provides an overview of the Naval Postgraduate School ARIES autonomous underwater vehicle and its guidance, navigation and control performance. An attempt is made to highlight its current operational capabilities and provide a description of future enhancements for greater mission utility and flexibility. An overview of the vehicle design along with descriptions of all major hardware components and sensors is given. A major discussion of the implementation of a modular, multi-rate, multi-process software architecture for ARIES autonomous control is provided. The architecture is designed to operate using either a single computer processor or two independent, cooperating processors linked through a network interface for improved load balancing. A dual computer implementation is presented here since each processor assumes different tasks for mission operation. Also included is a section on the underwater navigation method using a real-time extended Kalman filter that fuses all sensor data and computes the real time position, orientation, velocity, etc., of the vehicle. Experimental results for navigational accuracy using a DGPS / IMU / Doppler aided navigation system are presented with DGPS pop-up maneuvers.**

I. INTRODUCTION

The use of AUVs for Ocean Survey and military mine countermeasures is well documented (Curtin et. al., 1993, Smith et. al.,1998, and Allen, et. al., 97, among many others). The continuing problems facing further commercial use of such systems include accurate underwater navigation and communications links. Further research in low cost navigational accuracy and communications links is needed and toward that end, this paper presents a description of the latest generation of NPS underwater vehicle named the ARIES AUV. The ARIES vehicle is a shallow water communications server vehicle with a DGPS Doppler aided IMU / Compass navigation suite. Navigational errors are corrected by DGPS when surfaced, which, for shallow water applications presents no penalty. The vehicle is shown in a DGPS pop-up maneuver in Monterey Bay in Figure 1, and Figure 2 shows the component layout of the vehicle. The hull was outfitted in the fall of 1999 and has recently become fully operational (Spring 2000). The vehicle has been designed as a network server platform/target reacquisition vehicle, and has been operated during AUVFest '99 in Gulfport, MS (AUVFest '99).

Figure 1. The NPS ARIES AUV in a GPS Pop-up Maneuver (December 2000)

Report Documentation Page		*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE **2005**	2. REPORT TYPE	3. DATES COVERED **-**

4. TITLE AND SUBTITLE **Guidance and Control, For Small AUVs Using DGPS and Doppler Aided Inertial Underwater Navigation**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Naval Postgraduate School,Center for AUV Research,Monterey,CA,93943-5000**	8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT
This paper provides an overview of the Naval Postgraduate School ARIES autonomous underwater vehicle and its guidance, navigation and control performance. An attempt is made to highlight its' current operational capabilities and provide a description of future enhancements for greater mission utility and flexibility. An overview of the vehicle design along with descriptions of all major hardware components and sensors is given. A major discussion of the implementation of a modular, multi-rate, multi-process software architecture for ARIES autonomous control is provided. The architecture is designed to operate using either a single computer processor or two independent, cooperating processors linked through a network interface for improved load balancing. A dual computer implementation is presented here since each processor assumes different tasks for mission operation. Also included is a section on the underwater navigation method using a real-time extended Kalman filter that fuses all sensor data and computes the real time position, orientation, velocity, etc., of the vehicle. Experimental results for navigational accuracy using a DGPS / IMU / Doppler aided navigation system are presented with DGPS pop-up maneuvers.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**11**	

II. VEHICLE DESCRIPTION

Descriptions of the major hardware components of the ARIES shown in Figure 2, are given below.

Dimensions and Endurance: The vehicle weighs 225 Kg and measures approximately 3 m long, 0.4 m wide and 0.25 m high. The hull is constructed of ¼" thick 6061 aluminum and forms the main pressure vessel that houses all electronics, computers, and batteries. A flooded fiberglass nose is used to house the external sensors and power on/off switches and status indicators. It is capable of a top speed of 3.5 knots and is powered by six 12 volt rechargeable lead acid batteries. The endurance is approximately 4 hours at top speed, 20 hours hotel load only. The ARIES was primarily designed for shallow water operations and can operate safely down to 30 meters. However, with hull strengthening in certain areas, a depth of 100 meters may be attained.

Propulsion and Motion Control Systems: Main propulsion is achieved using twin ½ Hp electric drive thrusters located at the stern. During normal flight, heading and depth is controlled using upper bow and stern rudders and a set of bow planes and stern planes. Since the control fins are ineffective during very slow or zero forward speed maneuvers, vertical and lateral cross-body thrusters are to be used to control surge, sway, heave, pitch, and yaw, motions.

Navigation Sensors: The sensor suite used for navigation includes a 1200 kHz RD Instruments Navigator DVL that also contains a TCM2 magnetic compass. This instrument measures the vehicle ground speed, altitude, and magnetic heading. Angular rates and accelerations are measured using a Systron Donner 3-axis Motion Pak IMU that is considered to be a low cost "Tactical" grade IMU. While surfaced, carrier phase differential GPS (DGPS CP) is available to correct any navigational errors accumulated during the submerged phases of a mission. In addition, and because of inaccuracies in the TCM2 compass, a Honeywell HMR3000 magneto-restrictive compass, corrected by a deviation table, is used as the primary heading reference standard.
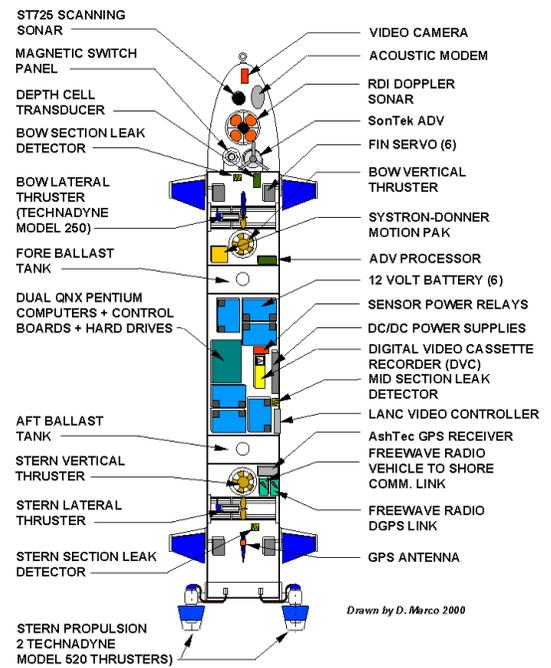
Figure 2. Hardware Components of the NPS ARIES

Sonar and Video Sensors: A Tritech ST725 scanning sonar or an ST1000 profiling sonar is used for obstacle avoidance and target acquisition/reacquisition. The sonar heads can scan continuously through 360° of rotation or swept through a defined angular sector. A fixed focus wide-angle video camera (Deep Sea Power and Light - SS100) is located in the nose and connected to a DV recorder. The computer is interfaced to the recorder and controls on/off and start/stop record functions. While recording, the date, time, vehicle position, depth and altitude is superimposed on the video image.

Vehicle/Operator Communications: Radio Modems are used for moderate bandwidth command, control (300 bytes/sec over 4 nm. with repeaters), and system monitoring while the vehicle is deployed and surfaced. While submerged, an acoustic modem will be used for low bandwidth communications. In the laboratory environment, a high-speed thin-wire ethernet connection is used for software development and mission data upload/download.

III. COMPUTER HARDWARE ARCHITECTURE

The dual computer system unit consists of two Ampro Little Board 166 MHz Pentium computers with 64 MB RAM, four serial ports, a network adapter, and a 2.5 GB hard drive each. Two DC/DC voltage converters for powering both computer systems and peripherals are integrated into the computer package. The entire computer system draws a nominal 48 Watts. Both systems use TCP/IP sockets over 'thin-wire' ethernet for inter processor communications and connections to an external LAN. The sensor data gathering computer is designated QNXT, while the second is named QNXE and executes the various auto-pilots for servo level control. Both computers are used as the baseboard for a stack of Diamond Systems PC-104 data acquisition boards.

IV. COMPUTER SOFTWARE ARCHITECTURE

A. The Architecture

A diagram outlining the modular, multi-rate, multi-process software architecture is shown in Figure 3. The architecture is designed to operate using a single computer processor or two, independent, co-operating processors linked through a network interface. Splitting the processing between two computers can significantly improve computational load balancing and software segregation. A dual computer implementation will be presented here, since in the ARIES, each processor assumes different tasks for mission operation. Both computers run the QNX real time operating system using synchronous socket sender and receiver network processes for data sharing between the two. Inter-process communication is achieved using semaphore controlled shared memory structures. At boot time, the network processes are started automatically and all shared memory segments are created in order to minimize the amount of manual setup performed by the user.

All vehicle sensors are interrogated by separate, independently controlled, **_concurrent_** processes, and there is no restriction on whether the processes operate **_synchronously_** or **_asynchronously_**. Since various sensors gather data at different rates, each process may be tailored to operate at the acquisition speed of the respective sensor. Each process may be started, stopped, or reset independently allowing easy reconfiguration of the sensor suite needed for a given mission. All processes are written in C.

To allow synchronous sensor fusion, each process contains a unique shared memory data structure that is updated at the specific rate of each sensor. All sensor data are accessible to a **_synchronous_** navigation process through shared memory and is a main feature of the software architecture. Incorporated into the navigation process is an extended Kalman filter that fuses all sensor data and computes the real time position, orientation, velocity, etc., of the vehicle. The dual computer implementation uses one processor for data gathering and running the navigation filters, while the second uses the output from the filters to operate the various auto-pilots for servo level control. Once the state information is computed, it is transmitted to the second computer over standard TCP/IP sockets.
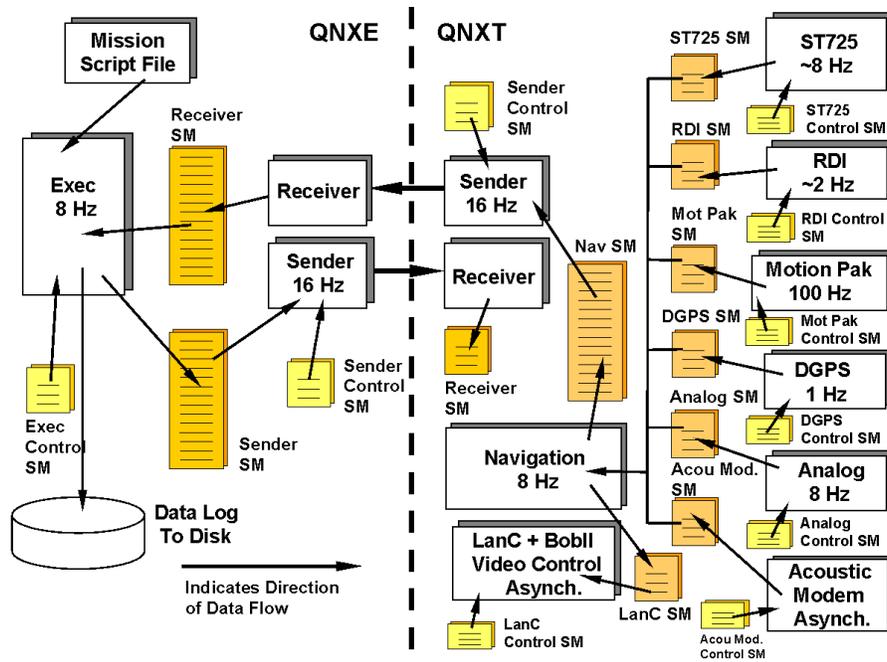
Figure 3. Dual Computer Software Architecture

B. Mission Control Modes

All vehicle behaviors are determined by a pre-programmed mission script file. This is parsed in the QNXE computer by the process *Exec*. The file contains a sequential list of commands that vehicle is to follow during a mission. These commands may be as simple as setting the stern propulsion thruster speeds to more complex maneuvers such as commanding the vehicle to repeatedly fly over a submerged target at a given GPS coordinate using altitude and cross track error control.

Below is an example of a simple mission script file.

| SET_ALTITUDE 2.0 |
| SET_HEADING 60.0 |
| SET_ STERN_THRUSTER_SPEED 700.0 |
| USE_ALTITUDE_CONTROL |
| USE_ HEADING_CONTROL |
| USE_STERN_THRUSTER_SPEED_CONTROL |
| SET_FLIGHT_DURATION 300.0 |

| SHUTDOWN |

This commands the vehicle to fly above the bottom at an altitude of 2 meters with a heading of 60 degrees, while running the twin stern thrusters at 700 rpm. The run is designed to last for a total of 300 seconds. The above mission could also have used depth instead of altitude control by simply replacing SET_ALTITUDE with SET_DEPTH and USE_ALTITUDE_CONTROL with USE_DEPTH_CONTROL.

V.ARIES AUTOPILOT CONTROL LAWS

The NPS ARIES currently uses four different autopilots for flight maneuvering control. They consist of independent diving, steering, altitude above bottom, and cross-track error controllers. All four autopilots are based on sliding mode control theory and each mode (i.e. diving, steering) is decoupled for ease of implementation and design. A reference for the details of controller design methodology may be found in (Healey and Lienard, 1993). These control laws are not unique, for example, fuzzy and heuristic

4

control is used in the FAU vehicles (An, et. al., 1997). However, the authors here have found that Sliding Mode controllers are simple to use and implement with minimal tuning.

A. Depth Controller

Since the vehicle depth can be independently controlled by the dive planes alone, the diving controller may be modeled by a linearized system with a single generalized input control, $u(t)$, generating a pitch-dive control distributed to bow and stern planes in an equal and opposite amount, and is of the form

$$\dot{x} = Ax + bu , \qquad (1)$$

and for the ARIES, the dynamics are given by the system of equations

$$\begin{bmatrix} \dot{q}(t) \\ \dot{\mathsf{q}}(t) \\ \dot{Z}(t) \end{bmatrix} = \begin{bmatrix} -1.3899 & -0.0032 & 0 \\ 1 & 0 & 0 \\ 0 & -U & 0 \end{bmatrix} \begin{bmatrix} q(t) \\ \mathsf{q}(t) \\ Z(t) \end{bmatrix} + \begin{bmatrix} -2.6091 \\ 0 \\ 0 \end{bmatrix} \mathsf{d}_{sp}(t) + (disturbances) \qquad (2)$$

where $q(t)$ is the pitch rate, $\mathsf{q}(t)$ is the pitch angle, $Z(t)$ is the depth in meters, and $\mathsf{d}_{sp}(t)$ is the stern plane angle in radians. U is the nominal longitudinal speed of the vehicle expressed in (m/sec) and a value of 1.8 m/sec is used. Although the bow and stern planes may be independently controlled, currently both sets of planes operate as coupled pairs such that the command to the bow planes is $-\mathsf{d}_{sp}(t)$. Notice that the heave velocity, w, equation is ignored, as also are its effects on the q and Z equations of motion. They are considered to be disturbances. The reduction of the system to third order creates a simplification that is both valid and useful.

The sliding surface is then formed as a linear combination of state variable errors in the usual way. Ignoring any nonzero pitch angle and rate commands, the sliding surface polynomial becomes

$$\acute{o}(t) = 0.7693q(t) + 0.6385\grave{e}(t) + 0.072488(Z_{com} - Z(t)) \qquad (3)$$

and the corresponding control law for the stern planes is

$$\ddot{a}_{sp}(t) = 0.4994(-0.4105q(t) + 0.1086\grave{e}(t) + \varsigma\, tanh(\mathsf{s}(t)/\mathsf{f}) \qquad (4)$$

where $\mathsf{h} = 1.0$ and $\mathsf{f} = 0.5$.

B. Altitude Controller

In order to control the vehicle altitude above the bottom designated $h(t)$, we simply need to change some of the signs of the terms from the diving equations. Noting the sign difference of the pitch angle and rate coefficients, this results in the following sliding surface

$$\acute{o}(t) = -0.7693q(t) - 0.6385\grave{e}(t) + 0.0724(h_{com} - h(t)) \cdot \qquad (5)$$

The stern plane command for altitude control is

$$\mathsf{d}_{sp}(t) = -0.4994(0.4105q(t) - 0.1086\mathsf{q}(t) + \mathsf{h}tanh(\mathsf{s}(t)/\mathsf{f})) \qquad (6)$$

where $\mathsf{h} = 1.0$, $\mathsf{f} = 0.5$, and $h(t)$ in meters replaces the vehicle depth, Z.

C. Heading Controller

By similar reasoning, and to eliminate the need to feedback the sideslip velocity, we argue that a second order model is sufficient. The sideslip effects are treated as disturbances that the control overcomes. Thus the heading model becomes

$$\dot{r}(t) = ar(t) + b\mathsf{d}_r(t) + (disturbances) \\ \dot{\mathsf{y}}(t) = r(t) \qquad (7)$$

where $r(t)$ is the yaw rate and $\mathsf{d}_r(t)$ is the stern rudder angle. The coefficients a and b have

been determined using system identification techniques from past in water experiments and are $a = -0.30\,rad/sec$ and $b = -0.1125\,rad/sec^2$. The stern and bow rudders operate in the same way as the planes therefore, the command to the bow rudder is $-\mathsf{d}_r(t)$.

Notice that in order to use this steering law, (ψ_{com}-ψ) must lie between $\pm180^0$, and is de-wrapped as needed in order to make that happen, and ignoring any nonzero command yaw rate, the sliding surface is defined by

$$\mathsf{s}(t) = -0.9499\,r(t) + 0.1701(\psi_{com}\text{-}\psi(t)).\tag{8}$$

The stern rudder command for heading control is

$$\mathsf{d}_r(t) = -1.543(2.5394r(t) + ç\,tanh(\mathsf{s}(t)/\mathsf{f}))\tag{9}$$

where $\mathsf{h} = 1.0$ and $\mathsf{f} = 0.5$.

D. Cross Track Error Controller

To follow a set of straight line tracks that form the basis of many guidance requirements, a sliding mode controller is presented that has been experimentally validated under a wide variety of conditions. Other works have studied this problem for land robots, (for example, Kanayama, 1990) and usually develop a stable guidance law based on cross track error. Here, with Figure 4 as a guide to the definitions, we use a combination of a line of sight guidance (Healey, Lienard, 1993) and a cross track error control for situations where the vehicle to track heading error is less than 40 degrees. For the line of sight guidance with large heading error, a separate line of sight controller is used.

One of the shortcomings of the heading controller defined above, is that it has no ability to track a straight line path between two way points, since it can only regulate the vehicle heading. It is desired to command the vehicle to track a line between two way-points with both a minimum of error from the track and heading error between the vehicle and the track. This type of regulation is known as cross track error control and the variable definitions are illustrated in Figure 4.
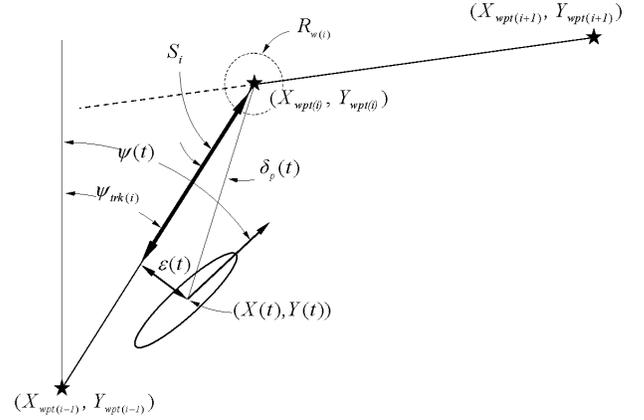

Figure 4. Cross Track Error Definitions

The variable of interest to minimize is the cross track error, $\mathsf{e}(t)$, and is defined as the perpendicular distance between the center of the vehicle (located at ($X(t), Y(t)$) and the adjacent track line. The total track length between way point i and $i\text{-}1$ is given by

$$L_i = \sqrt{(X_{wpt(i)} - X_{wpt(i-1)})^2 + (Y_{wpt(i)} - Y_{wpt(i-1)})^2},\tag{10}$$

where the ordered pairs ($X_{wpt(i)}, Y_{wpt(i)}$) and ($Y_{wpt(i-1)}, X_{wpt(i-1)}$) are the current and previous way points respectively. The track angle, $\psi_{trk(i)}$, is defined by

$$\psi_{trk(i)} = tan2^{-1}(Y_{wpt(i)} - Y_{wpt(i-1)}, X_{wpt(i)} - X_{wpt(i-1)}).\tag{11}$$

The cross track heading error $\tilde{\psi}(t)_{CTE(i)}$ for the ith segment is defined as

6

$$\widetilde{y}(t)_{CTE(i)} = y(t) - y_{trk(i)} \qquad (12)$$

where $\widetilde{y}(t)_{CTE(i)}$ must be normalized to lie between $\pm 180^0$. The difference between the current vehicle position and the next way point is

$$\widetilde{X}(t)_{wpt(i)} = X_{wpt(i)} - X(t)$$
$$\widetilde{Y}(t)_{wpt(i)} = Y_{wpt(i)} - Y(t) \qquad (13)$$

With the above definitions, the distance to the ith way point projected to the track line $S(t)_i$, can be calculated using

$$S(t)_i = \widetilde{X}(t)_{wpt(i)}(X_{wpt(i)} - X_{wpt(i-1)}) + \widetilde{Y}(t)_{wpt(i)}(Y_{wpt(i)} - Y_{wpt(i-1)})/L_i \qquad (14)$$

therefore, $S(t)_i$ ranges from 0-100% of L_i.

The cross track error $e(t)$ may now be defined as

$$e(t) = S_i(t) \sin(\ddot{a}_p(t)) \qquad (15)$$

where $\ddot{a}_p(t)$ is the angle between the line of sight to the next way point and the current track line given by

$$\ddot{a}_p(t) = \tan 2^{-1}(Y_{wpt(i)} - Y_{wpt(i-1)}, X_{wpt(i)} - X_{wpt(i-1)})$$
$$- \tan 2^{-1}(\widetilde{Y}(t)_{wpt(i)} - \widetilde{X}(t)_{wpt(i)}) \qquad (16)$$

and must be normalized to lie between $\pm 180^0$.

With the cross track error defined, the sliding surface can be cast in terms of derivatives of the errors such that

$$e(t) = e(t)$$
$$\dot{e}(t) = U \sin(\widetilde{y}(t)_{CTE(i)})$$
$$\ddot{e}(t) = Ur(t) \cos(\widetilde{y}(t)_{CTE(i)})$$
$$\dddot{e}(t) = U\dot{r}(t) \cos(\widetilde{y}(t)_{CTE(i)}) - Ur(t)^2 \sin(\widetilde{y}(t)_{CTE(i)})$$

The sliding surface for the cross track error controller becomes a second order polynomial of the form

$$s(t) = \ddot{e}(t) + l_1 \dot{e}(t) + l_2 e(t) \qquad (17)$$

The condition for stability of the sliding mode controller is

$$\dot{s}(t) = \dddot{e}(t) + l_1 \ddot{e}(t) + l_2 \dot{e}(t) = -h(s/f), \qquad (18)$$

and to recover the input for control, the heading dynamics Equation (7) may be substituted into Equation (16) to obtain

$$U(ar(t) + bd_r)\cos(\widetilde{y}(t)_{CTE(i)}) - Ur(t)^2 \sin(\widetilde{y}(t)_{CTE(i)}) + \cdots$$
$$l_1 Ur(t)\cos(\widetilde{y}(t)_{CTE(i)}) + l_2 U \sin(\widetilde{y}(t)_{CTE(i)}) \qquad (19)$$

Rewriting Equation (15), the sliding surface becomes

$$s(t) = Ur(t)\cos(\widetilde{\sigma}(t)_{CTE(i)}) + l_1 U \sin(\widetilde{\sigma}(t)_{CTE(i)}) + l_2 e(\qquad (20)$$

The rudder input can be expressed as

$$d_r(t) = \left(\frac{1}{Ub\cos(\widetilde{\sigma}(t)_{CTE(i)})}\right)\left(-Uar(t)\cos(\widetilde{\sigma}(t)_{CTE(i)})\right.$$
$$+ U(r(t))^2 \sin(\widetilde{\sigma}(t)_{CTE(i)}) - l_1 Ur(t)\cos(\widetilde{\sigma}(t)_{CTE(i)})$$
$$\left. - l_2 U \sin(\widetilde{\sigma}(t)_{CTE(i)}) - \varsigma(s(t)/f)\right) \qquad (21)$$

where $l_1 = 0.6$, $l_2 = 0.1$, $h = 0.1$, and $f = 0.5$. To avoid division by zero, in the rare case where $\cos(\widetilde{\sigma}(t)_{CTE}) = 0.0$ (i.e. the vehicle heading is perpendicular to the track line) the rudder command is set to zero since this condition is transient in nature.

Line of Sight Controller

When the condition arises that the magnitude of the cross track heading error $|\widetilde{\sigma}(t)_{CTE(i)}|$ exceeds 40^0, a Line of Sight Control (LOS) is

used. In this situation, the heading command can be determined from

$$\psi(t)_{com(LOS)} = \tan 2^{-1}(\widetilde{Y}(t)_{wpt(i)}, \widetilde{X}(t)_{wpt(i)}) \tag{22}$$

and the LOS error from

$$\widetilde{\psi}(t)_{LOS} = \psi(t)_{com(LOS)} - \psi(t), \tag{23}$$

and the control laws used for heading control, Equations (8,9) may be used.

Two conditions may be true for the way point index to be incremented. The first and most usual case is if the vehicle has penetrated the way point watch radius $R_{w(i)}$. Secondly if a large amount of cross track error is present, the next way point will become active if the projected distance to the way point $S(t)_i$ reached some minimum value $S_{min(i)}$, such that

$$if \left(\sqrt{(\widetilde{X}(t)_{wpt(i)})^2 + (\widetilde{Y}(t)_{wpt(i)})^2} \leq R_{w(i)} \ || \ S(t)_i < S_{min(i)} \right) \ \text{THEN}$$

Activate Next Way Point

In water experimental results using the controllers presented above will now be presented in the next section.

VI. NAVIGATION

The ARIES vehicle uses an INS / DOPPLER / DGPS navigational suite and an Extended Kalman Filter (EKF) which was developed and presented in (Healey, An, Marco, 1998), and may be tuned for optimal performance given a set of data. The main impediments to navigational accuracy are the heading reference and the speed over ground measurement. In this system, the heading reference is derived from both the Honeywell compass and the Systron Donner IMU, which provides yaw rate. The fusion of the yaw rate and the compass data leads to an identification of the yaw rate bias, which is assumed to be a constant value. The compass bias which is mostly dependent on vehicle heading relative to magnetic north (An, 1997) is identified in the EKF (Healey, An,1998)

(Grenon, 2001) using DGPS positions when surfaced. When submerged, the position error covariance grows, but is corrected on surfacing. A relatively short surface time, (for example, 10 seconds) allows the filter to re-estimate biases, correct position estimates and continue with improved accuracy. As a demonstration, the ARIES vehicle was operated in Monterey Bay, in a series of runs including a dive-surface-dive-surface sequence. Figure 5, below shows a plot of vehicle position in an exercise where the vehicle is commanded to follow a track at depth, come up for a DGPS correction, then follow the bottom at an altitude of 3m, while a video is recorded from a down-looking camera. The vehicle then surfaces to get a second fix before turning round and repeating the exercise from the complementary heading. In this plot, the vehicle trajectory is designed to fly over the Monterey Inner Shelf Observatory (MISO) Instrument Frame placed in 12 meters of water approximately 0.5 kilometers from shore with estimated GPS position used to design the approach lane. The video taken as the vehicle flies over the MISO is designed to provide identification details of the arbitrary object given its approximate DGPS location point.
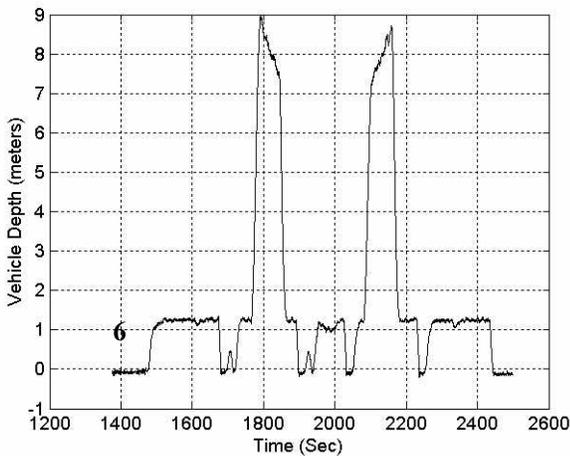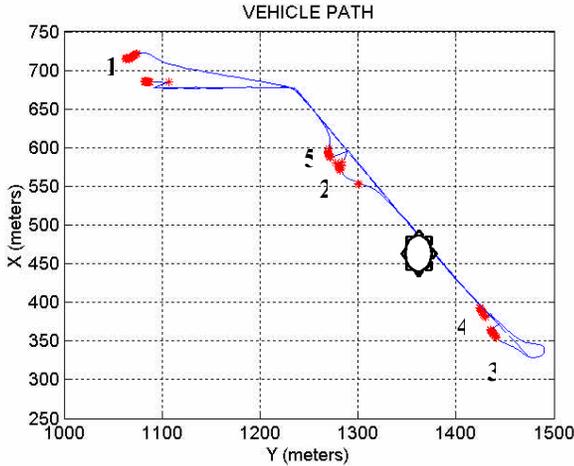
Figure 5a. Vehicle Path showing locations where the GPS position fixes were obtained by surfacing for 20 seconds (asterisks). 5b. Depth response during run that clearly shows the DGPS pop up maneuvers.
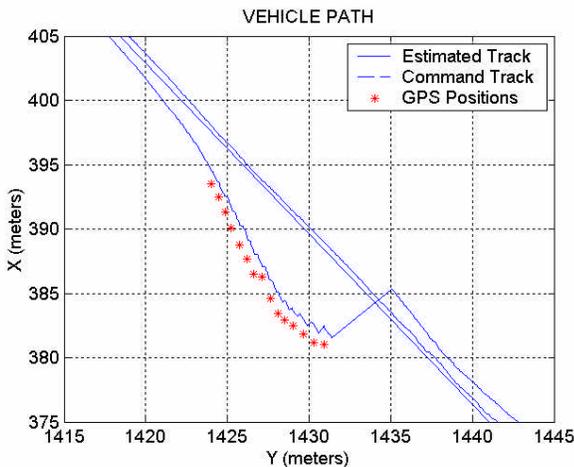

Figure 6. Close up of the Final Surface Showing the Filter Solution together with the DGPS Measurement At the Surface.

In Figure 6, a close up of the final surfacing maneuver shows that there is great consistency in estimating the true DGPS data point as seen by the AshTec G-12 unit on board. The difference between the Kalman Filter solution and the DGPS data points while surfaced is sub meter precision. However, the difference between the dead reckoning solution underwater is a few meters off the mark.

In Figure 7, the number of visible satellite vehicles seen by the DGPS unit are shown to evolve quickly. Within 10 seconds, 9 satellites are being used to compute the position solution.
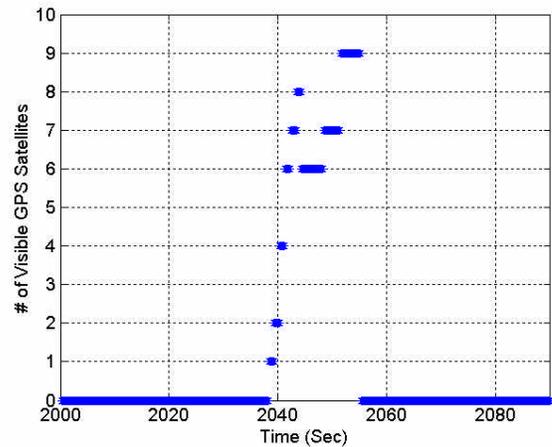

Figure 7. Time history of the Response of the Number of Visible GPS Satellites during the Surface phase shown in Figure 6.
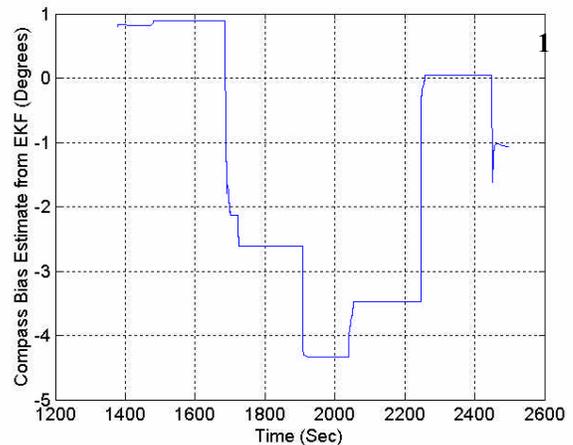

Figure 8. Compass Bias Estimate versus time.

9

Figure 8 shows the response of the heading bias estimate from the EKF for the entire run. At each surface approximately 10 DGPS points are obtained which rapidly corrects the compass bias. However, as is seen, compass corrections in the neighborhood of 5 degrees are still needed to predict correctly the vehicle positions. This is an indication that further corrections of the compass deviation table are needed. The remaining question is whether or not the deviations are predictable or random. While some additional runs suggest that there may be some degree of consistency, it remains to be shown conclusively.

In spite of the above, the navigation accuracy was sufficient to identify using video the MISO Laboratory 4 times out of 5 passes. The photo in Figure 9 shows an image of the structure from 3 m above bottom.

Figure 9 Photo of MISO frame structure from 3 m above bottom in 12 m Water Depth

VII. SERVER VEHICLE CONCEPT

It is proposed to use the NPS ARIES as a network communications and navigation server vehicle for multi-vehicle cooperative operations. One of the needs is underwater data transfer between network nodes through noisy communication channels. Use of the server vehicle as a data relay increases the range of communications of the underwater components of the network. Figure 9 describes the concept of operations where the ARIES communicates through an aerial relay to act as a command and control server for other vehicles. Its acoustic modem would be used with multiple worker vehicles that are engaged in a search pattern. Figure 9 below shows the ARIES on the surface using a radio modem to report mission status of the worker vehicles (possibly vehicle positions, image snippets of targets, and hydrographic data) to the command ship. Navy assets in mine hunting use the MEDAL software system as a common operating environment for the viewing of asset positions, contact reports and bathymetry data.
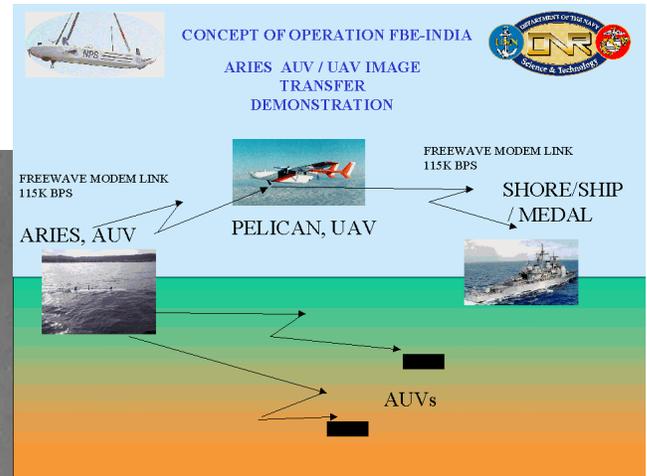

Figure 9. Sever vehicle concept. Low bandwidth submerged data transfer between underwater vehicles. High-speed radio data relay to and from command ship.

VIII. CONCLUSIONS

This paper has described a third generation AUV from the NPS Center for AUV Research. A new computer architecture has been described to enable the vehicle to operate as a network server using acoustic and radio communication links. Most importantly, the vehicle has been designed for accurate navigation in shallow water using an extended Kalman filter and DGPS-CP. In spite of many efforts to generate an accurate compass deviation table with less than one degree of error, heading errors larger than this, appear to remain. Position errors less

than 5 meters can be found using a second DGPS correction in most cases. To obtain position errors within 1 meter, extreme care in developing the compass deviation table must be taken. In the future to get errors within 1 % of distance traveled will require a high quality, high cost, navigation grade IMU. The results have shown that the control errors are very small and all controllers are well behaved and 8 to 9 GPS satellites can be acquired with 10 seconds of surfacing. Research is ongoing in the field of multi-vehicle cooperative behavior and common control languages.

IX. REFERENCES

Allen, B., Stokey, R., Austin, T., et. al., "REMUS: A Small Low Cost AUV: System Description, Field Trials, Performance Results", *Proceedings IEEE Oceans 97*, pp. 994-1000, 1997

An, P. E., Healey, A. J., Park, J., Smith, S. M., " Asynchronous Data Fusion For AUV Navigation Via Heuristic Fuzzy Filtering Techniques ", *Proceedings IEEE, Oceans 97*, Halifax, Oct. 1997 IEEE CD-ROM 0-7803-4111-2 http://web.nps.navy.mil/~me/healey/papers/oceans_97.pdf

AUVFest '99, NAVO Report http://www.cnmoc.navy.mil/ auvdemo/index.htm

Curtin, T, Bellingham, J., Catapovic, J, Webb,D., "Autonomous Oceanographic Sampling Network" *Oceanography*, V6, N3, pp. 86-94,1993

Grennon, G., An, E. P., Smith, S.M., Healey, A. J., "Enhancement of the Inertial Navigation System for the Morpheous Autonomous Underwater Vehicles", *IEEE Journal of Oceanic Engineering*, v26, n. 4., pp 548-560.

Healey, A. J., An, E. A., Marco, D. B., " On Line Compensation of Heading Sensor Bias for Low Cost AUV's", *Proceedings of the IEEE Workshop on Autonomous Underwater Vehicles*, AUV98, IEEE Catalog. Number 98CH36290, ISBN # 0-7803-5190-8, August 20-21, 1998, Cambridge, Mass. pp 35-42 http://web.nps.navy.mil/~me/healey/papers/auv98.pdf

Healey, A.J., Lienard, D. "Multivariable Sliding Mode Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles'*, IEEE Journal of Oceanic Engineering*, Vol. 114, No.3, July, 1993

Kanayama, Y., Hartmann, B. I., "Smooth Local Path Planning for Autonomous Vehicles" *Autonomous Robot Vehicles*, Springer-Verlag, ISBN 000-387-97240-4, pp.-62-68, 1990.

Smith, S. M., Ganesean, K., An, E.A., Dunn, S., "Strategies for Simultaneous Multiple AUV Operation and Control," *International Journal of System Science*, V29, N10, pp. 1045-1063,1998

X ACKNOWLEDGEMENTS